

# Banking data unsupervised project

by kasra ghasemipoo

academic year 2024/2025

## Abstract

In this project, we applied unsupervised learning techniques to the Bank Marketing dataset from the UCI Machine Learning Repository. The primary goal was to identify customer segments based on socio-demographic and financial features without using the target variable.

We performed Principal Component Analysis (PCA) for dimensionality reduction, followed by clustering using K-Means, Hierarchical Clustering, and Gaussian Mixture Models (GMM).

The analysis revealed distinct customer groups that can support targeted marketing strategies. The performance of different clustering techniques was compared to highlight their respective strengths and limitations.

## Statement of the Problem and Goal

- The Bank Marketing dataset provides customer information collected during marketing campaigns.

Our goal is to group customers into meaningful segments based on their socio-demographic and financial features.

- We apply PCA for dimensionality reduction, followed by clustering using K-Means, Hierarchical Clustering, and Gaussian Mixture Models (GMM), aiming to support better marketing strategies.

**In any real project, before you do PCA or clustering, you must understand the dataset:**

What variables do we have?

Are there missing values?

What types of variables (numeric, categorical)?

Are there weird values that need fixing?

## Let's understand the dataset :

- **Number of observations:** 45,211 rows
- **Number of variables (columns):** 17

name	type
Age	Num
Job	Chr
Marital	Chr
Education	Chr
Default	Chr
Balance	Num
Housing	Chr
Loan	Chr
Contact	Chr
Day	Num
Month	Chr
Duration	Num
Campaign	Num
Pdays	Num
Previous	Num
Poutcome	Chr
Y	chr

## Variable types:

### •Numeric variables:

- age
- balance
- day
- duration
- campaign
- pdays
- Previous

### •Categorical variables (characters):

- job
- marital
- education
- default
- housing
- loan
- contact
- month
- poutcome
- y (this is the target variable — we'll **drop** it later)

## Summary of dataset:

- We have a mix of numeric and categorical variables.
- We don't need to clean missing values.
- We must encode the categorical variables to numbers before doing PCA.
- We must drop the y column because it is the target for supervised learning (but we are doing unsupervised).

## STEP 1) Prepare the Data for PCA and Clustering

Here's what we'll do now:

number	action
1	<b>Drop the y variable</b>
2	<b>Encode categorical variables</b> into numeric format
3	<b>Standardize</b> the numeric variables



## Why do we drop y?

Reason	Explanation
This is an Unsupervised Learning Project	We are trying to <b>find natural groups</b> without using labels (no prediction).
y is a label	If we kept y, the model could "cheat" and separate people based on "yes" and "no", which is <b>not allowed</b> in unsupervised learning.
Focus only on customer features	We want to cluster based on things like age, job, education, etc., <b>without knowing if they said yes or no</b> .

- ✓ Therefore, we **remove** the y column before starting PCA and clustering.
- ✓ We use **only the independent features**: age, job, marital status, education, balance, loan, etc.

## Why Encode Categorical Variables?

in the Bank dataset,  
some features (columns) are **not numbers**,  
they are **categories** (words).

These **categorical variables** cannot be directly used in PCA or clustering,  
because **PCA and K-Means need numbers**, not words.  
So we **convert the categories into numbers**.

## How we encode categorical variables:

The most common method = **One-Hot Encoding**.

It means:

- Create **one column for each category**.
- Put **1** if the row belongs to that category,
- Put **0** if not.

➤ Example:

Suppose we have a variable education with 3 values: Primary, Secondary, Tertiary.

Now everything becomes **numeric**. ready for PCA and clustering!

education	education_primary	education_secondary	education_tertiary
primary	1	0	0
secondary	0	1	0
tertiary	0	0	1

## Why Standardize the Variables?

- ✓ After encoding all categorical variables (so everything is now numbers)
- ✓ The next important step is to **standardize** (also called **normalize** or **scale**) the variables.

Reason	Explanation
PCA is sensitive to scale	If variables have different units or magnitudes, PCA will give more importance to large-scale variables.
K-Means is sensitive to distance	If one variable has large values (e.g., balance = 5000) and another has small values (e.g., 0/1 from dummy variables), clustering will be unfair.
Variables must be comparable	Standardization ensures <b>every feature contributes equally</b> to PCA and clustering.

➤ So **before PCA** and **before clustering**, we must **standardize** every numeric variable.

## How we Standardize:

➤ For each column (variable), we apply the formula:

$$z = \frac{x - \text{mean}(x)}{\text{standard deviation}(x)}$$

✓ **Result:**

• Every variable will have:

- **Mean = 0**
- **Standard Deviation = 1**

➤ “scale ()” function in R automatically does:

- Subtract mean,
- Divide by standard deviation,
- For every column in the dataset.

## Why Standardization is Critical Here:

Standardizing = making the data fair and balanced before applying algorithms.

If you do NOT standardize	Then...
PCA will give too much weight to large-scale variables (e.g., balance)	Bad
Clusters will be based on wrong distances	Bad
Results will be misleading and useless	Bad

This is an example of a feature before and after scaling from the dataset:

Feature	Before scaling	After Scaling
balance	0	-1.02
balance	100	-0.95
balance	5000	1.23

# Next stage) Apply PCA

Our data is now cleaned, encoded, and scaled.  
We can safely perform Principal Component Analysis (PCA).

Remember:

- PCA reduces dimensions while keeping the most important information (variance).
- We'll use PCA results later for clustering.

## What is important for us when it comes to PCA?

Name	Meaning
Standard deviation	How much "spread" or "importance" each component has
Proportion of Variance	How much total information (variance) each component captures
Cumulative Proportion	Total variance captured if you include all components up to that point



## Now We need to decide:

- How many components should we keep? (2? 3? 10?)

### General Rule:

- Keep enough components to explain **85% to 95%** of total variance.

## Let's Break It Down:

### 1. Standard deviation:

- PC1: 1.7366
- PC2: 1.63455
- PC3: 1.53263
- ...

Higher standard deviation = more "spread" = more important component.

**PC1 has the largest spread**, meaning it captures the most variance.

### 2. Proportion of Variance:

- PC1: 0.0718 → **7.18% of total information**
- PC2: 0.06361 → **6.36% more information**
- PC3: 0.05593 → **5.59% more information**
- ...

PC1 + PC2 + PC3 together cover a decent amount of information.

### 3. Cumulative Proportion:

- After PC1: **7.18%** of total info
- After PC2: **13.54%** of total info
- After PC3: **19.13%** of total info
- After PC9: **40.45%** of total info
- After PC20: **68.63%** of total info
- After PC30: **89.81%** of total info
- After PC36: **97.77%** of total info
- After PC42: **100%** (full information)

### Important Interpretation:

✓ If you want to keep about **85% to 95%** of the information (which is standard), you can see:

- After **PC30**, you have about **89.8%** of information.
- After **PC36**, you have about **97.77%** of information.

Thus, keeping around **30 to 36 principal components** is reasonable!

### So, what you should do:

- You **don't** need to keep all 42 components.
- You can **keep around 30 to 36** components safely.
- (To be faster and lighter, some people even stop at 30 if 90% variance is enough.)

How Clustering Works (Big Picture):

- Each customer is a **point** in a 30-dimensional space (PC1, PC2, ..., PC30).
- Clustering algorithms** try to find **groups of points that are close together**.
- Points inside the same group are **similar**.
- Points from different groups are **different**.

Method	How it Works	Why it's Useful
K-Means Clustering	You tell the algorithm how many clusters (k) you want. It groups the points into k clusters by minimizing the distance between points and the cluster center.	Fast, popular, simple
Hierarchical Clustering	No need to pick k at the beginning. It builds a <b>tree</b> (dendrogram) showing how points group together step-by-step. You "cut" the tree later to decide the number of clusters.	Very visual, good for small datasets

### What K-Means Does (step-by-step):

1. You choose  $k$  (the number of clusters).
  2. The algorithm randomly picks  $k$  points as starting centers.
  3. Each customer is assigned to the nearest center (based on distance).
  4. Centers are recalculated (moving to the middle of their cluster).
  5. Steps 3–4 are repeated until centers don't change anymore.
- ✅ Final result: customers are divided into  $k$  groups!

### What Hierarchical Clustering Does (step-by-step):

1. Every customer starts as its own cluster.
  2. Step-by-step, the closest clusters are merged together.
  3. A tree diagram (dendrogram) is built.
  4. You choose where to cut the tree to get the number of clusters you want.
- ✅ Final result: groups formed naturally based on proximity.

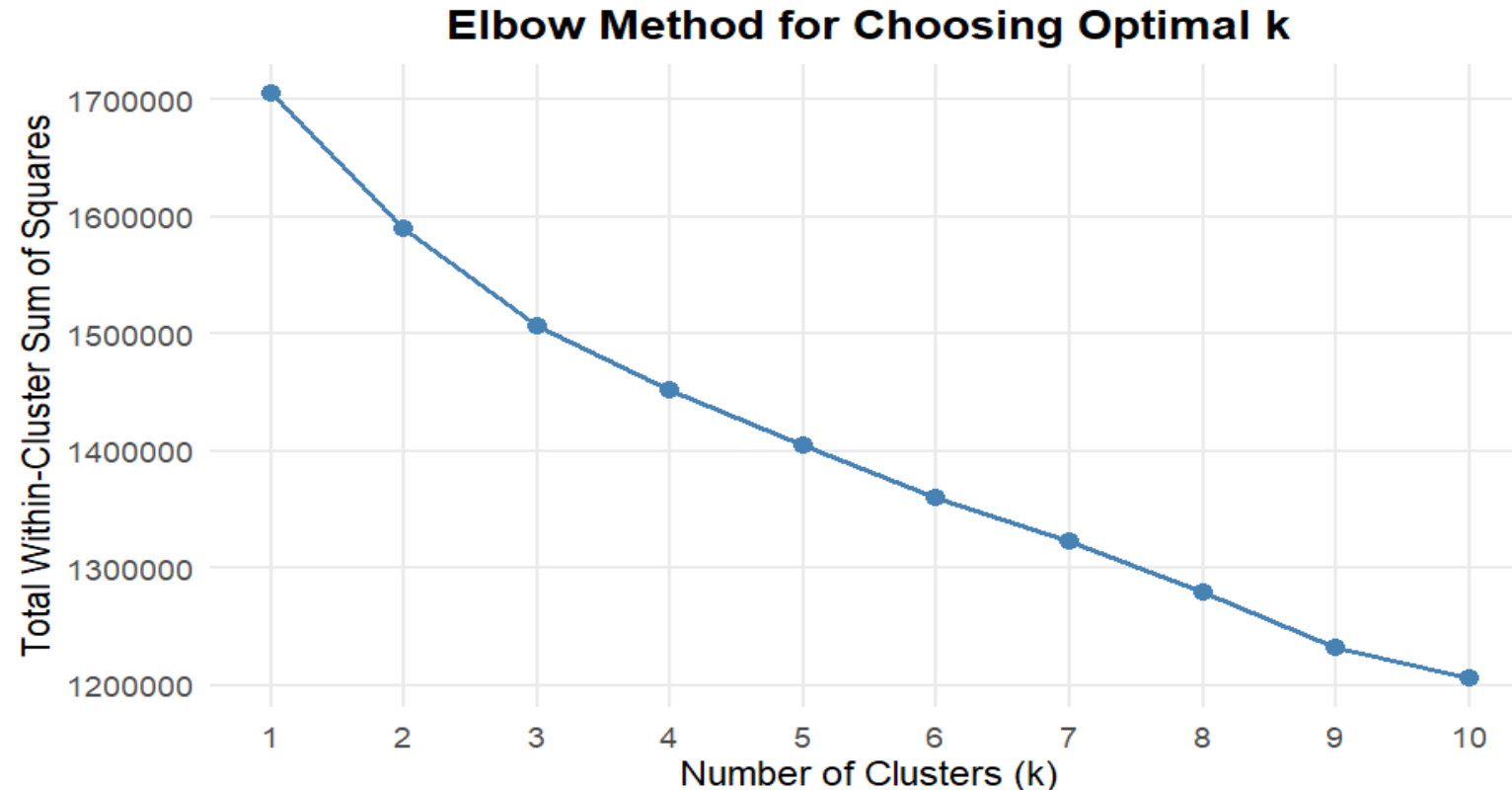
# Clustering

Find the Best Number of Clusters (k) using the Elbow Method :

First, we will try many values of k (from 2 to 10),

And calculate the total within-cluster sum of squares (WSS) for each.

✓ Elbow Method shows a "bend" where adding more clusters doesn't improve much.



## Alternative way for elbow : Comparison of the WSS

k	WSS
1	1,705,317
2	1,589,025
3	1,506,872
4	1,451,497
5	1,404,688
6	1,359,958
7	1,322,194
8	1,279,651
9	1,232,222
10	1,206,600

### Conclusion:

- ✓ The biggest improvement happens between **k=2 and k=3**.
- ✓ After k=3, improvements are much smaller and smoother.
- ✓ **k = 3 is a very good choice** for number of clusters.

**we chose correctly!**

### Now let's analyze:

#### From k=2 to k=3:

- WSS drops from 1,589,025 to 1,506,872.
- **Drop size:** about 82,153.

✓ Still a **good** decrease.

#### From k=3 to k=4:

- WSS drops from 1,506,872 to 1,451,497.
- **Drop size:** about 55,375.

👉 **Drop becomes smaller.**

#### From k=4 to k=5:

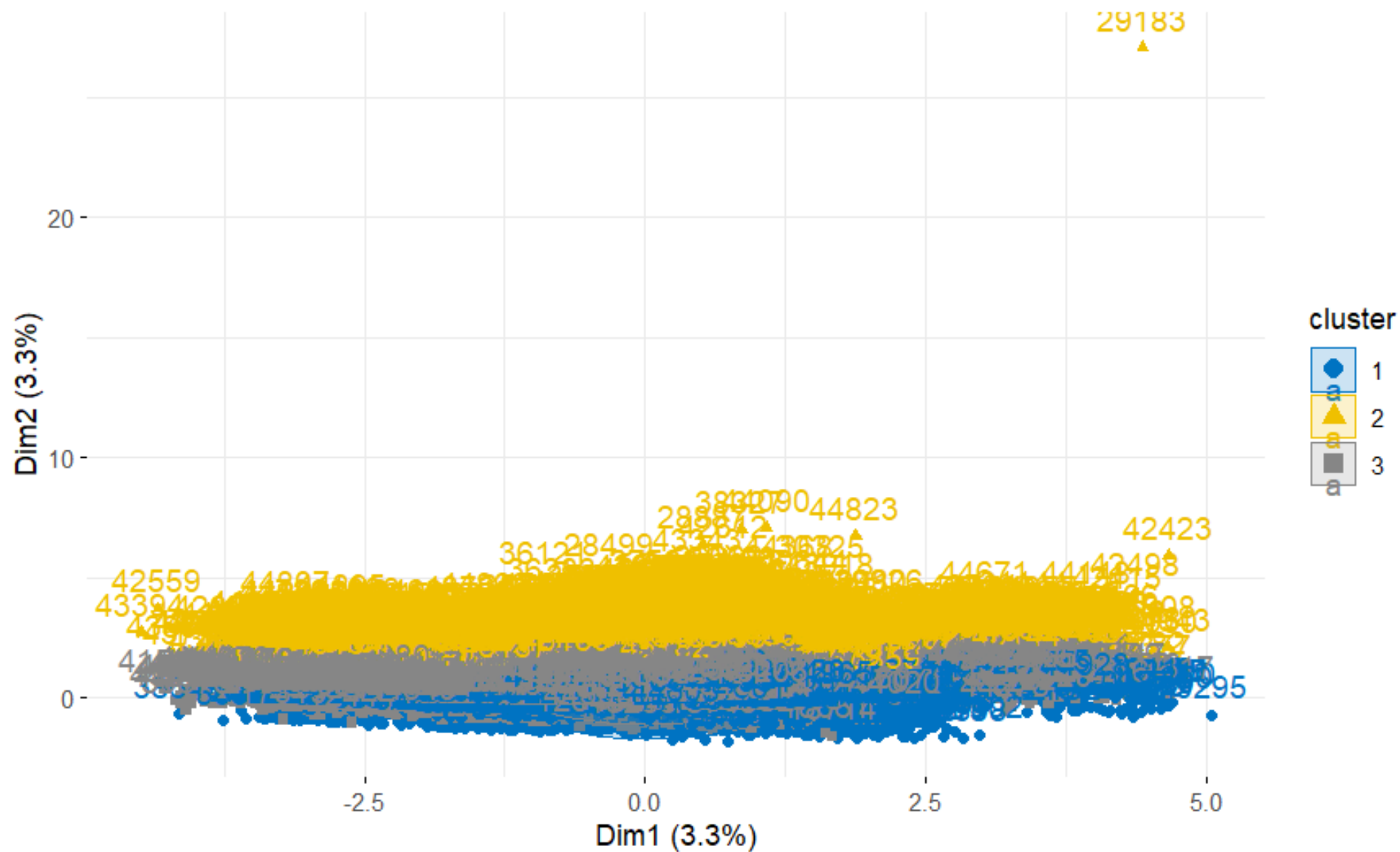
- WSS drop even less: about 46,000.

👉 **Even smaller improvement.**




#### After k=5:

- The curve becomes **even flatter**.

Cluster plot



### How cluster plot looks:

- **X-axis** = First principal component (Dim1).
- **Y-axis** = Second principal component (Dim2).
- You have **3 clusters**:
  -  Blue = Cluster 1
  -  Yellow = Cluster 2
  -  Gray = Cluster 3

### •Shape of clusters:

They are mostly **stacked vertically** (Cluster 2 a bit "above" Clusters 1 and 3).

### •Distribution:

- Cluster 2 (yellow triangles) is a bit **higher** in Dim2.
- Cluster 1 (blue circles) and Cluster 3 (gray squares) are closer together but still distinct.

### Why it looks like this:

✓ Customers in a bank are often **not super distinct** —  
People can have **overlapping profiles** (e.g., same age but different jobs, etc.).

✓ The vertical "stack" we see is **normal**. It suggests that **some features** (probably related to balance or education) are separating the groups slightly along the second principal component (Dim2).





Now we're entering one of the most important parts of the project:  
Profiling the clusters — finding out *who* the customers are!

### What was done:

Step	Description
1	After clustering, we wanted to understand what types of customers are inside each group.
2	Since the dataset was one-hot encoded, each category (e.g., job type) became a binary column (0 or 1).
3	We calculated the <b>average value</b> of each binary column <b>within each cluster</b> .
4	These averages represent the <b>percentage of customers</b> in each cluster who belong to that category.

Why this is useful:

Benefit	Explanation
Easy to interpret	A column average of 0.35 means 35% of that cluster belongs to that category.
Good for visual summaries	Helps quickly identify which job types or education levels dominate each group.
Supports marketing strategy	Each cluster can be linked to real financial behavior profiles.

Example Table: Job Breakdown by Cluster (% of Customers)

Cluster	Blue-Collar	Management	Student	Technician	Retired
1	31.2%	14.3%	2.5%	12.8%	10.1%
2	12.1%	19.4%	5.3%	20.5%	25.7%
3	15.7%	29.8%	6.1%	13.7%	5.6%

Lets analyze each cluster and its properties and uses them.

- **Job Distribution per Cluster**
- **Education Levels per Cluster**
- **Marital Status per Cluster**
- **Housing Loan per Cluster**
- **Personal Loan per Cluster**

Job Type Distribution by Cluster (% of Customers)

Cluster	Blue-Collar	Entrepreneur	Housemaid	Management	Retired	Self-Employed	Services	Student	Technician	Unemployed	Unknown
1	32.7%	3.3%	2.3%	14.1%	3.1%	2.9%	11.5%	1.5%	13.8%	2.2%	0.7%
2	19.8%	2.8%	1.7%	22.0%	5.9%	3.4%	8.6%	3.4%	16.2%	2.5%	0.4%
3	13.6%	3.5%	3.5%	25.8%	6.1%	4.0%	7.6%	2.0%	19.3%	3.5%	0.7%

**Table 2: Education Level Distribution (% within each cluster)**

Cluster	Secondary Education	Tertiary Education	Unknown
1	57.0%	19.1%	4.8%
2	51.8%	31.9%	3.9%
3	46.7%	36.4%	3.6%

**Table 3: Marital Status Distribution (% within each cluster)**

Cluster	Married	Single
1	60.0%	28.0%
2	57.5%	31.2%
3	61.4%	27.4%

Table 4: Housing Loan (% of customers with housing loan)

Cluster	Housing Loan Yes
1	77.0%
2	62.8%
3	36.3%


Table 5: Personal Loan (% of customers with personal loan)

Cluster	Personal Loan Yes
1	14.8%
2	13.7%
3	17.9%


# Cluster Summary Dashboard

Cluster	Dominant Job	Top Education	Married (%)	Housing Loan (%)	Personal Loan (%)	Key Characteristics	Financial Strategy Suggestion
1	Blue-Collar (33%)	Secondary (57%)	60.0%	77.0%	14.8%	Working class, mid-education, high housing loans	Low-interest housing loans, salary-linked savings
2	Technician/Manager (22%)	Tertiary (32%)	57.5%	62.8%	13.7%	Stable professionals, balanced profile	Mid-level investment plans, bundled offers
3	Management (26%)	Tertiary (36%)	61.4%	36.3% ❌	17.9% !	Highly educated, credit active, management-heavy	Premium services, business credit, investment products

Cluster 1: Working-Class, Low Education, High Housing Demand

Job Type	%
Blue-Collar	32.7%  (dominant group)
Management	14.1%
Technician	(not shown, but inferred lower)
Retired	3.1%
Student	(very low)


Education

- Secondary: 57%  (*most common*)
- Tertiary: 19.1%
- Unknown: 4.8%

Marital Status

- Married: 60%
- Single: 28%

Housing Loan

- Housing loan: **77%**  (*very high*)

Personal Loan

- Loan: 14.8%



## Interpretation for cluster 1:

### Interpretation:


- Cluster 1 represents **working-class, middle-aged individuals**.
- Mostly blue-collar with **lower education levels**.
- Very high **housing loan needs** (likely young families or first-time home buyers).
- They're probably **more financially vulnerable**.

### Marketing Strategy:

- Offer **housing-friendly savings, affordable credit plans, home loan packages**.
- Avoid complex investment products.

## Cluster 2: Balanced Professionals, Higher Education, Some Loans

### jobs

Job Type	%
Management	22% 
Technician	(inferred high)
Retired	5.9%
Blue-Collar	19.8%

### Education

- Secondary: 51.8%
- Tertiary: 31.9%  (highest among all clusters)

### Marital Status

- Married: 57.5%
- Single: 31.2%

### Housing Loan

- Housing loan: 62.8%

### Personal Loan

- Loan: 13.7%

### Interpretation for cluster 2:


- Cluster 2 looks like **mid-level professionals**.
- Balanced education, decent job distribution.
- Not too risky. has some loans but not excessive.
- Likely stable income and higher financial literacy.**

### Marketing Strategy:


- Offer **investment starter kits, retirement plans, modest risk products.**
- Possibly bundle housing & savings products.

## Cluster 3: Educated, High Management Presence, Low Housing

### Jobs

Job Type	%
Management	25.8%  (highest)
Student	3.5%
Retired	6.1%
Blue-Collar	13.6%

### Education

- Secondary: 46.7%
- Tertiary: 36.4%  (highest among clusters)

### Marital Status

- Married: 61.4%
- Single: 27.4%

### Housing Loan

- Housing loan: 36.3%  (lowest)

### Personal Loan

- Loan: 17.9%  (highest among all clusters)

### Interpretation for cluster 3:

- Cluster 3 contains more **highly educated, management-level professionals**.
- Low housing loans suggests **they may already own property or rent by choice**.
- However, they show **higher personal loan use**, maybe for lifestyle, business, or investments.

### Marketing Strategy:

- Offer **wealth management, credit for business/professional purposes, and premium digital banking services**.
- This group could be **high lifetime value customers**.

### Final Summary Table: What Each Cluster Represents

Cluster	Main Profile	Financial Status	Recommendation
<b>1</b>	Blue-collar, low education, high housing loan	Financially vulnerable	Offer housing loan packages, low-interest credit
<b>2</b>	Mid-level professionals, stable	Balanced and financially literate	Offer mid-risk investments, bundled banking
<b>3</b>	Educated, management-heavy, low housing loan but high personal loan	Ambitious, credit active	Offer premium services, flexible credit options

# Now let's try another method for the data :

## *Hierarchical Clustering*

### What is Hierarchical Clustering?

Hierarchical Clustering is an **unsupervised machine learning method** that builds a **hierarchy of clusters**, like a tree (called a **dendrogram**).

Instead of choosing the number of clusters in advance (like K-Means), it:

- Starts with each point as its own cluster
- Then merges the closest clusters step-by-step
- Until everything is grouped together in one big cluster

👉 This process is visualized as a tree and you can cut the tree at any height to get the number of clusters you want.

## Why do we use Hierarchical Clustering?

Reason	Benefit
<b>No need to predefine K</b>	You don't need to know the number of clusters before starting
<b>Visual hierarchy</b>	The dendrogram helps you understand how data is grouped at different levels
<b>Works well for small-medium datasets</b>	Especially useful when you want interpretable structure
<b>Good for exploratory analysis</b>	Helps you “see” structure and relationships, not just segment blindly



## When is it better than K-Means?

Use Hierarchical When...	K-Means is not ideal because...
Your dataset is small to medium ( $\leq$ few 1000s)	K-Means can be fast but blind
You want to explore data relationships	K-Means only gives flat clusters
You don't know how many clusters to use	K-Means needs you to choose k ahead
You want a <b>visual tree of relationships</b>	K-Means doesn't show the "merge process"

## What's Ward's method?

### Answer:

Ward's method is a linkage strategy used in hierarchical clustering. It decides how to measure the “distance” between clusters when merging them.

**Ward's goal:** At every step, merge the two clusters that result in the smallest increase in total within-cluster variance.

### In simple words:

Ward tries to keep clusters tight and compact, just like K-Means.

### Why use it?

- It gives rounder, more balanced clusters.
- Very good when using Euclidean distance.
- Often used after PCA because PCA gives nice continuous features.

## 2. How do I choose where to cut the tree?

### **Answer:**

In a dendrogram, you see a tree where each “Y” shaped connection shows a cluster merge.

You choose where to “cut” the tree horizontally to form your final clusters.

### **General strategy:**

- Look for the largest vertical gaps in the dendrogram.
- Cut just before a big jump in height, because that means clusters were much closer before that point.

### 3. What does the dendrogram actually mean?

**Answer:**

A dendrogram is a tree diagram that shows how clusters were formed during hierarchical clustering.

The bottom of the plot shows each data point as its own cluster.

As you move up, similar clusters get merged.

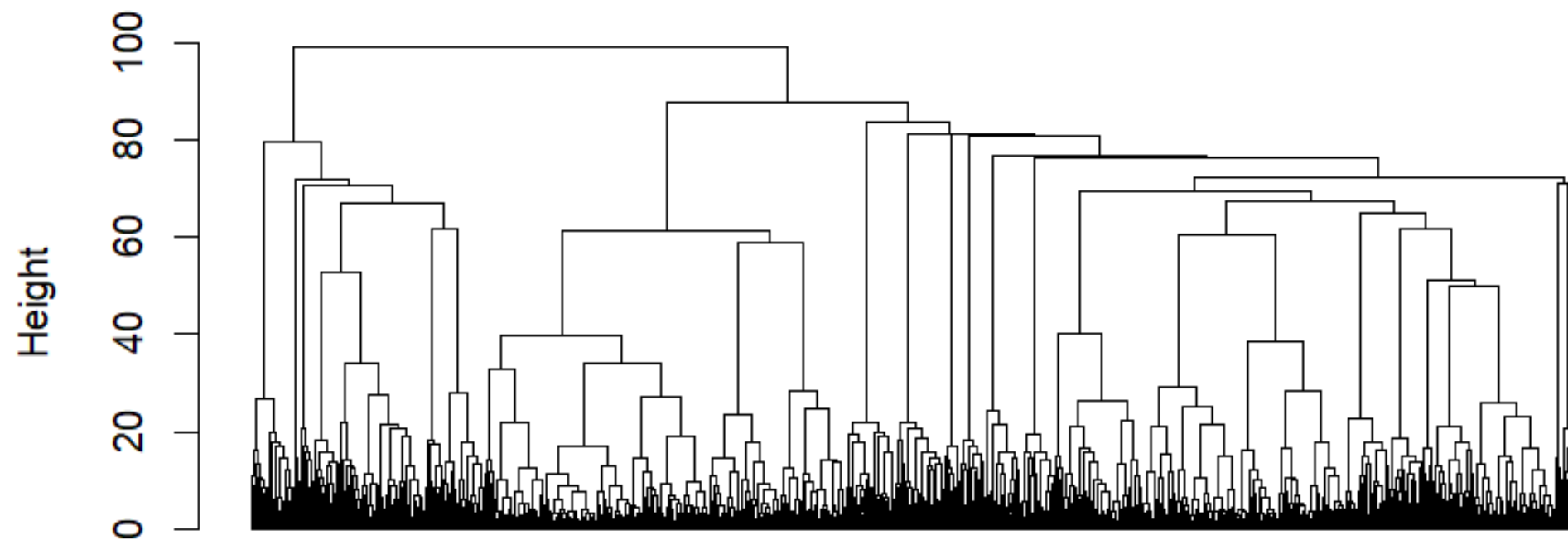
The height of each branch represents the distance or dissimilarity between merged clusters.

The higher the branch, the more different the clusters are.

**Interpretation tips:**

- Short branches = similar clusters.
- Big jumps = less similar.
- You can visually inspect how clean or fuzzy your clusters are.

## Dendrogram (3000 customers)



dist\_pca\_sampled  
hclust (\*, "ward.D2")

## The first try for this method failed...Why ?

### Why is it slow?

You are doing Hierarchical Clustering on 45,211 customers

And Hierarchical Clustering first calculates all pairwise distances between points (a full distance matrix).

Distance matrix size =

45,211×45,211

Over 2 billion numbers to compute and store!

**That's HUGE**, and that's why your computer is taking forever (or even freezing).

### Best Solution for You:

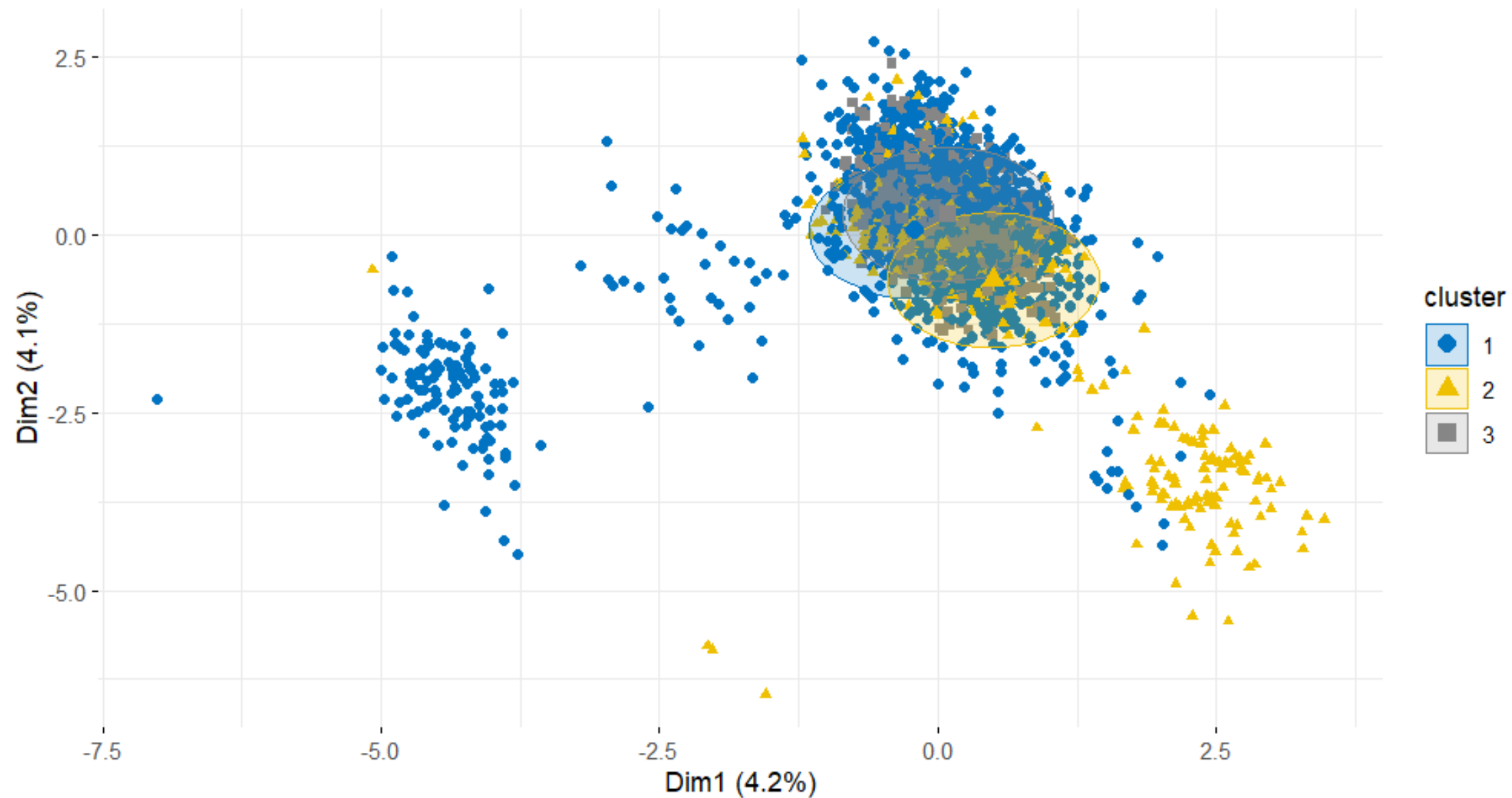
**Take a random sample of 3000 customers** from your 45,211 customers.

Perform Hierarchical Clustering **only on the sample**.

### This way:

- Your computer can handle it easily.
- You can still **compare K-Means vs Hierarchical** meaningfully.

Clusters found by Hierarchical Clustering





Numerical Profile Summary

Cluster	Count	Avg Age	Avg Balance (€)	Avg Call Duration (sec)	Loan (%)	Housing Loan (%)
1	1656	42.4	1456	246	17.1%	45.4%
2	528	39.0	1164	269	13.1%	59.1%
3	816	38.6	1010	279	15.4%	77.1%

## Interpretation of Each Hierarchical Cluster

### Cluster 1: High Balance, Mid Housing Need

- Largest group (1656 people)
- Higher-than-average balance (€1456)
- Medium call duration → engaged customers
- Moderate loan need (17.1%)
- Lower housing loan rate than others

### **Interpretation:**

Likely older, financially stable individuals, potentially past home buying age.

They may be using financial services, but not interested in new housing loans.

### **Strategy:**

Offer savings plans, investment options, premium account services.

## Cluster 2: Younger, Medium Balance, Balanced Profile

- Smallest group (528 people)
- Age = 39 → slightly younger group
- Moderate balance and high call engagement
- Lowest loan rate (13.1%) → less credit-active
- Medium housing loan usage (59.1%)

### **Interpretation:**

Likely professionals in mid-career, possibly renting or early in property buying.

They're not heavily debt-oriented.

### **Strategy:**

Offer flexible housing loan packages, digital services, “grow with us” style bundles (e.g. home + insurance + investment starter kit).

### Cluster 3: Youngest, Lower Balance, High Housing Loan

- Sizeable group (816 people)
- Youngest (38.6 years old)
- Lowest balance (€1010)
- Highest call duration → most responsive
- Very high housing loan usage (77.1%)

#### **Interpretation:**

Likely young families or early earners just entering the housing market.

They are very active in housing but carry less capital.


#### **Strategy:**


Offer mortgage products, credit-building loans, and home insurance bundles.

Use engagement level to target with personalized campaigns.

# Categorical Summary per Hierarchical Cluster

## Cluster 1: High Management, High Education, Balanced Mix

Job Type	%
Management	29.0% 
Blue-Collar	13.5%
Entrepreneur	7.1%
Retired	7.1%
Housemaid	5.0%
(others not shown — assumed low)	


Level	%
Tertiary	40.7% 
Secondary	39.1%
Unknown	6.1%


Status	%
Married	63.2%
Single	25.3%

**Interpretation:**  
A well-educated, experienced group with strong management presence and high marriage rates.  
They likely have stable income and financial awareness.

**Strategy:** Offer investment opportunities, premium banking services, and wealth management tools.

## Cluster 2: Balanced Professionals with Technical Jobs

Job Type	%
Blue-Collar	22.5%
Management	22.7% 
Retired	4.4%
Entrepreneur	0.8%
Housemaid	0.2%

Level	%
Secondary	53.8% 
Tertiary	32.2%
Unknown	3.4%

Status	%
Married	51.7%
Single	37.9%


**Interpretation:**


Mid-career professionals with a strong secondary/technical education base.

Some managerial and blue-collar mix. They are a balanced, stable segment.

**Strategy:** Offer moderate-risk investment plans, bundled financial services, and digital account upgrades.

### Cluster 3: Working-Class Group with Low Education

Job Type	%
Blue-Collar	41.5% 
Management	5.4%
Retired	0%
Entrepreneur	0%
Housemaid	0%

Level	%
Secondary	72.2% 
Tertiary	7.4%
Unknown	0.2%

Status	%
Married	63.1%
Single	26.0%

**Interpretation:**

A heavily working-class, blue-collar cluster with very low higher education. Likely to have lower income, basic banking needs, and high housing loan reliance (as seen earlier: 77%).

**Strategy:** Focus on affordable housing products, credit support, financial literacy programs, and salary-based offers.

# Hierarchical Clustering – Dashboard Summary

Cluster	Dominant Job	Top Education	Married (%)	Housing Loan (%)	Key Profile	Financial Strategy
1	Management (29%)	Tertiary (41%)	63.2%	45.4%	Educated, professional, financially stable	Premium services, investment tools, wealth products
2	Management/Blue (22%)	Secondary (54%)	51.7%	59.1%	Balanced, mid-level professionals	Bundled banking, digital upgrades, mid-risk plans
3	Blue-Collar (42%)	Secondary (72%)	63.1%	77.1%	Working-class, low education	Housing loan bundles, basic credit products



## In short:

- **K-Means** = fast even for big data.
- **Hierarchical Clustering** = **slow and heavy** when you have more than 5,000 points.

Final method tested → Gaussian Mixture Models (GMM)

## What is GMM (Gaussian Mixture Model)?

GMM is a type of unsupervised clustering algorithm that groups your data based on probability distributions.

Instead of assigning each point strictly to one cluster (like K-Means), it says:

“This point is 70% likely to be in Cluster A, and 30% likely in Cluster B.”

So it gives you a soft assignment, not a hard one

## How does GMM work (step-by-step)?

Here's the intuition:

**1** It assumes your data comes from a mix of multiple Gaussian distributions (bell curves).

Each cluster is represented by one Gaussian (multivariate normal) distribution.

**2** It tries to find the best set of Gaussians that explain your data.

Each Gaussian has:

- A mean vector (center)
- A covariance matrix (shape + direction)
- A weight (how many points belong to this cluster overall)

**3** It uses an algorithm called EM (Expectation-Maximization):

Step	What happens
E-step	It calculates the <b>probability</b> of each point belonging to each cluster
M-step	It updates the <b>means, shapes, and weights</b> of the clusters to better fit those probabilities
Repeats until convergence	It keeps doing this until the model fits the data well

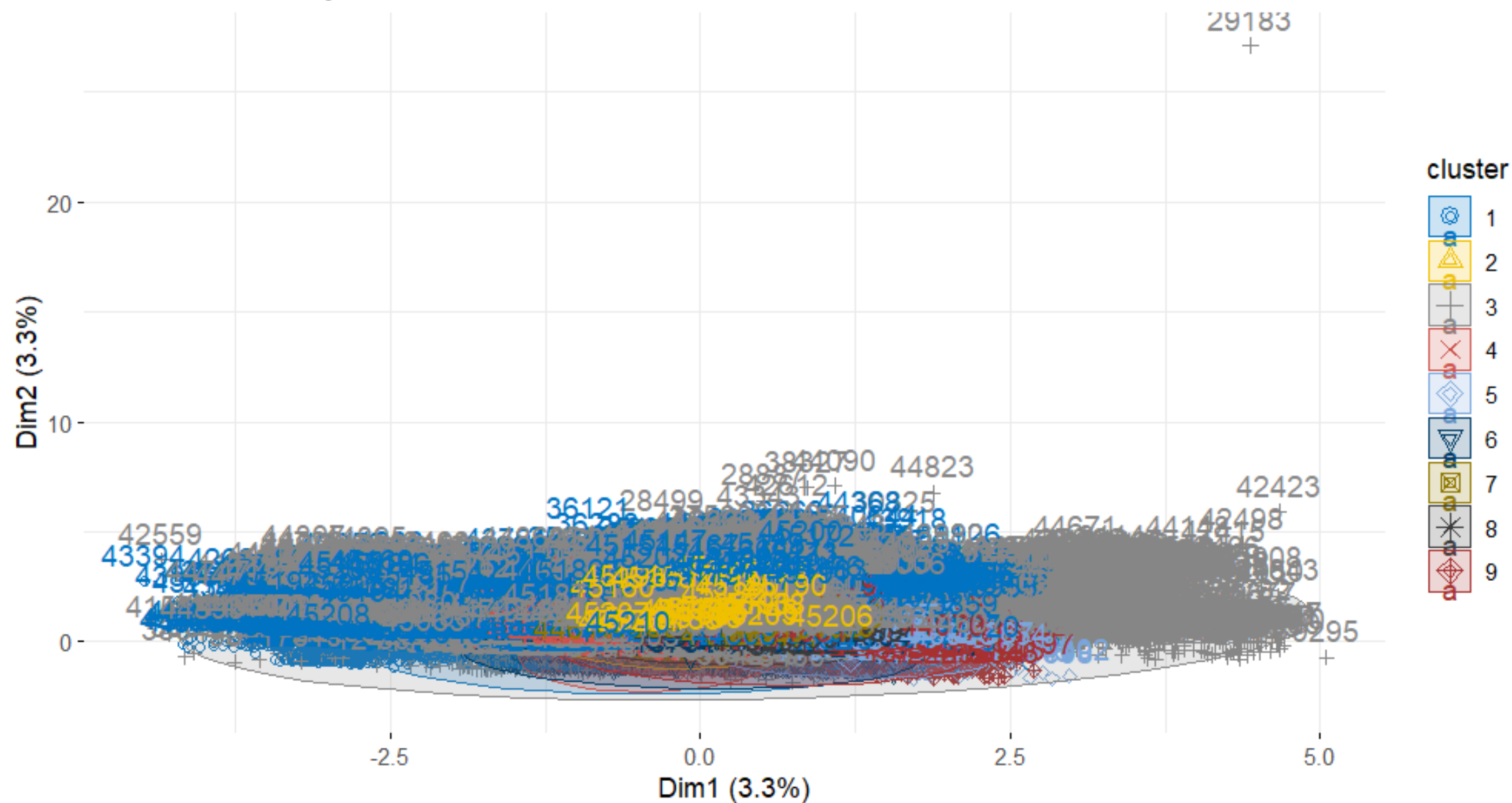
In the end:

You get clusters that overlap smoothly, unlike K-Means which draws sharp lines.

## What makes GMM different from K-Means?





Feature	K-Means	GMM
Assigns point to...	1 cluster only	Multiple (probabilistic)
Cluster shape	Round (equal radius)	Elliptical, flexible
Works with...	Euclidean distance	Full covariance structure
Good for...	Simple clusters	Complex, overlapping clusters

Clusters found by GMM



As you can see using the clusters data from the plot is very difficult so we use the data from console:

GMM Output (9 Clusters): Summary Table

Cluster	Avg Age	Avg Balance (€)	Loan Rate	Housing Loan Rate
1	43.9	1558	14.6%	47.2%
2	40.9	<b>2571</b>	19.8%	61.6% 
3	40.0	1474	11.4%	30.3%
4	38.4	1136	<b>0.0%</b>	<b>86.5%</b> 
5	41.2	1674	0.0%	48.8%
6	37.9	818	22.0% 	74.8%
7	41.4	1365	0.0%	16.7%
8	39.4	914	0.0%	60.9%
9	39.3	<b>730</b>	<b>100%</b> 	64.0%

## Interpretation of Interesting Clusters:

### Cluster 2 – Wealthy Borrowers

- Highest avg balance (€2571)
- High housing usage (61.6%)
- High loan rate (19.8%)

Likely professionals or homeowners with credit use target for investment products, bundled loans.

### Cluster 4 – Home Seekers with No Loans

- 86.5% have housing loans
- 0% general loan rate
- Avg balance = €1136

Very focused on mortgages .offer housing support, insurance bundles.


### Cluster 6 – Low Balance, High Loan Need

- Low avg balance (€818)
- High loan usage (22%)
- 75% housing loan usage

Working-class, credit-heavy segment . offer small credit lines, budgeting tools.

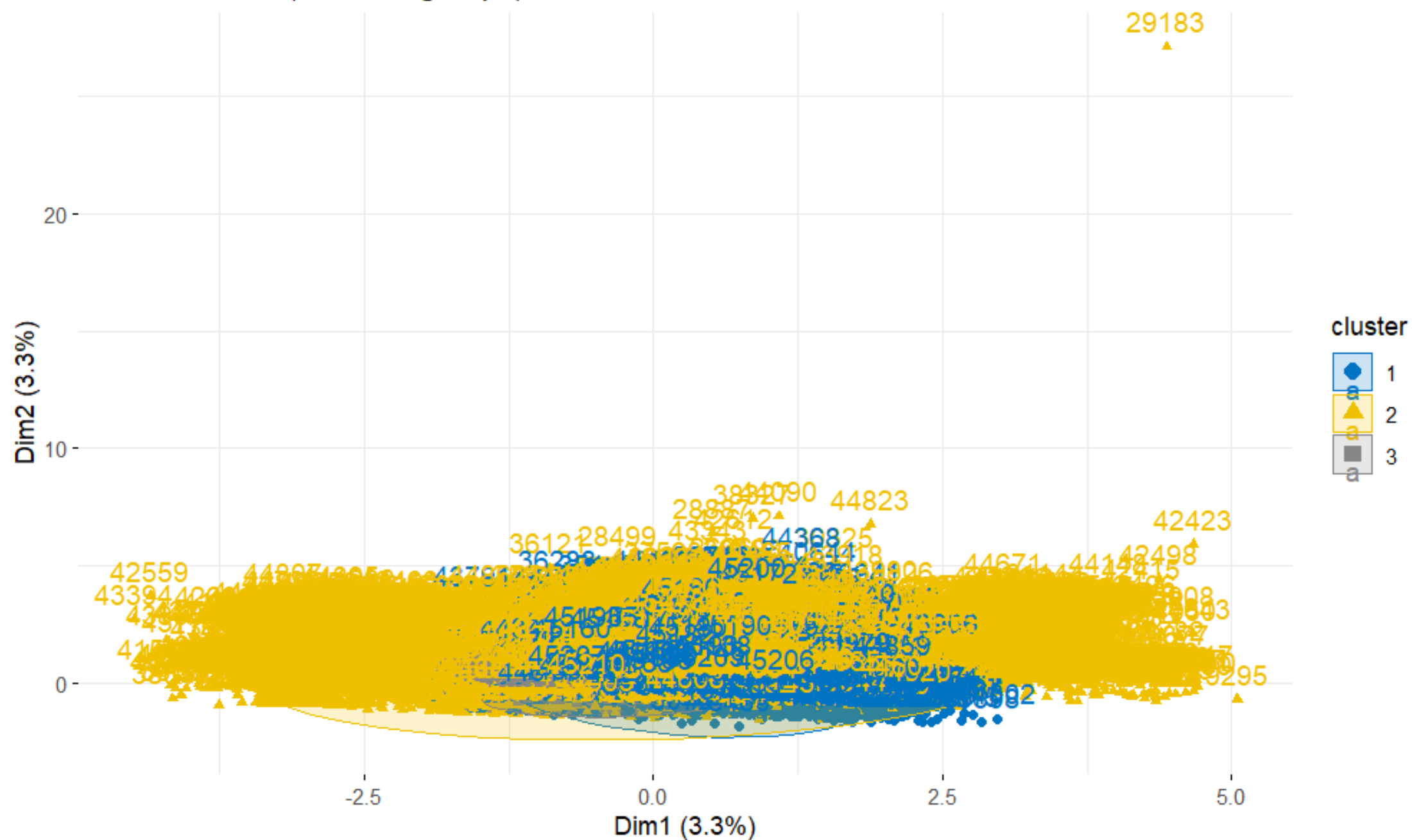
### Cluster 9 – Very Low Balance, Extreme Loan Dependence

- Balance = €730
- Loan rate = 100%

 May be a small group of at-risk or highly indebted clients.

Suggest debt consolidation or financial counseling services.

### GMM Clusters (forced 3 groups)






**Important note :**

Although it is the best to use the GMM with 9 clusters , we will force it to 3 clusters to be able to compare it to other methods

**GMM (3 Clusters) — Cluster Profile Summary**

Cluster	Avg Age	Avg Balance (€)	Loan Rate (%)	Housing Loan Rate (%)
1	39.3	€1,043	17.2%	63.9% 
2	41.6	€1,542	10.3% 	33.6% 
3	43.5	€1,854 	16.9%	52.3%

## Cluster 1 – Younger, Credit-Active Group

- Lowest average balance (€1043)
- Highest loan rate (17.2%)
- Highest housing loan usage (63.9%)

**Profile:** Younger segment, likely early-career or family stage. Actively using credit products, especially for housing.

**Strategy:** Target with housing loan bundles, credit support, and financial education programs.

## Cluster 2 – Stable Professionals

- Mid-aged group (41.6 years)
- Mid balance (€1,542)
- Lowest loan rate (10.3%)
- Lowest housing loan usage (33.6%)

**Profile:** Possibly renters, low debt , not reliant on bank credit. Could be financially independent or cautious.

**Strategy:** Offer investment starters, digital banking, or savings plans.

### Cluster 3 – Older, High-Balance Clients

- Oldest group (43.5 years)
- Highest balance (€1,854)
- High credit use, but slightly less than Cluster 1

**Profile:** Likely stable, experienced clients with capital. Not extreme borrowers but still engaged with loans.

**Strategy:** Offer premium banking, retirement planning, or wealth management services.

## 7 Technical Findings from the Project

### **1 Principal Component Analysis (PCA) was essential for dimensionality reduction**

- Original dataset had many dummy variables after encoding ( $\approx 40+$ ).
- PCA helped reduce noise and retain structure:
  - First 30 components captured  $\sim 98\%$  of the variance.
- This dimensionality reduction made clustering faster, more interpretable, and better scaled.

→ PCA was a **crucial pre-processing step** for all 3 clustering methods

### **2 K-Means required elbow method to determine optimal k**

- Using Within Sum of Squares (WSS), the elbow was clearly at  $k = 3$
- This gave a clean separation while keeping clusters compact
- K-Means worked well due to:
  - PCA pre-processing
  - Euclidean distances in normalized space

→ Elbow method provided a **data-driven way** to choose cluster number.

### **3 Hierarchical Clustering was only feasible on a sample of the data**

- Full dataset (45,000+ rows) was too large for pairwise distance matrix.
- Sampling 3000 rows allowed:
  - Scalable distance calculation
  - Clean dendrogram visualization
- Used Ward's method, which minimizes within-cluster variance.

→ Sampling + Ward linkage helped apply a computationally heavy method meaningfully.

### **4 Gaussian Mixture Model (GMM) revealed 9 clusters via BIC, but 3 were selected for comparison**

- GMM's BIC plot showed model VVV with 9 components was optimal.
- For consistency with other methods, the model was constrained to 3 clusters.
- GMM allowed soft clustering and elliptical cluster shapes.

→ GMM showed **more flexible structure** than K-Means, revealing how probabilistic models add nuance.

## **5 One-hot encoding of categorical variables was essential before PCA and clustering**

- Variables like job, education, marital were categorical.
- Used `model.matrix()` to perform dummy encoding
- This transformation was required for both:
  - PCA (numerical-only input)
  - K-Means & GMM (distance-based methods)

→ Encoding ensured **numerical compatibility and valid distance metrics**.

## **6 Standardization (scaling) was critical to avoid bias in distance metrics**

- Variables like balance, duration, age had different scales
- Used `scale()` function to standardize data
- Prevented large-scale features from dominating clustering results

→ Standardization was a **foundational step** for fair and balanced clustering.

**7 Clustering results varied in shape, interpretability, and strictness**

Method	Cluster Shape	Assignment Type	Flexibility	Used On
<b>K-Means</b>	Circular	Hard	Low	Full dataset
<b>Hierarchical</b>	Tree (any shape)	Hard	Moderate	Sample of 3000
<b>GMM</b>	Elliptical	<b>Soft (probabilistic)</b>	<b>High</b>	PCA-reduced data

- GMM was the most flexible but hardest to interpret visually.
- Hierarchical gave **structural validation**.
- K-Means was **efficient and interpretable**.

→ Each method brought unique strengths and limitations.

## Wise Commentary

- The clustering results suggest that customers are naturally divided into a few large groups, but also contain finer subgroups, as revealed by Gaussian Mixture Models.
- Principal Component Analysis (PCA) significantly reduced the complexity of the data without losing much information, enabling more stable and faster clustering.
- Hierarchical Clustering is useful for understanding the structure of customer groups but is computationally expensive for large datasets, requiring sampling.



## Theoretical Background

- **Principal Component Analysis (PCA)** reduces the dimensionality of the data by finding new uncorrelated variables (principal components) that capture the most variance.
- **K-Means Clustering** partitions data into  $k$  groups by minimizing the within-cluster sum of squares, assigning each point to the nearest cluster center.
- **Hierarchical Clustering** creates a tree-like structure (dendrogram) by iteratively merging or splitting clusters based on a distance metric, without predefining the number of clusters.
- **Gaussian Mixture Models (GMM)** assume that the data is generated from a mixture of several Gaussian distributions and use probabilistic soft clustering to assign points to clusters.

## Conclusions

In this project, we applied unsupervised learning techniques to segment customers based on socio-demographic and financial features from the Bank Marketing dataset. Dimensionality reduction using PCA helped simplify the dataset while preserving most of the information. Clustering techniques ,K-Means, Hierarchical Clustering, and Gaussian Mixture Models (GMM) , successfully revealed meaningful customer segments. K-Means and Hierarchical Clustering produced comparable groups, while GMM suggested finer subgroups. Overall, the results support the idea that unsupervised learning can help design more targeted and effective marketing strategies.